IEEE TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS

Multiobjective Identification of Controlling Areas in Neuronal Networks

Yang Tang, *Member, IEEE,* Huijun Gao, *Senior Member, IEEE,* and Jürgen Kurths

Abstract—In this paper, we investigate the multiobjective identification of controlling areas in the neuronal network of cats' brain by considering two measures of controllability simultaneously. By utilizing nondominated sorting mechanisms and composite differential evolution (CoDE), a reference point based nondominated sorting composite differential evolution (RP-NSCDE) is developed to tackle the multiobjective identification of controlling areas in the neuronal network. The proposed RP-NSCDE shows its promising performance in terms of accuracy and convergence speed, in comparison to nondominated sorting genetic algorithms II. The proposed method is also compared with other representative statistical methods in complex network theory, single objective and constraint optimization methods to illustrate its effectiveness and reliability. It is shown that there exists a trade-off between minimizing two objectives, and therefore pareto fronts (PFs) can be plotted. The developed approaches and findings can also be applied to coordination control of various kinds of real-world complex networks including biological networks and social networks, etc.

Index Terms—Synchronization, Neuronal networks, Controlling areas, Multiobjective optimization.

◆

1 INTRODUCTION

Most biological, social, and technological networks display complex topological features and dynamics, which can be modeled as complex networks [1]–[3]. Ideas originating from network science, have been successfully applied to the analysis of genetic regulatory networks [4]–[7], the study of high-level information processing of neuronal networks [8]–[12], and other kinds of practical biological networks. In particular, by employing the information of structural and functional MRI, diffusion tensor imaging, magnetoencephalography and electroencephalography and recent methods of complex network theory, the brain's structural and functional systems have been shown a spatial topology and typical features of complex networks, such as the existence of highly connected hubs, small-world topology, and modularity-both at a whole-brain and a cellular scale [10]. It was also shown that the organization of brain networks display an economical trade-off between the physical cost of brain networks and the adaptive value of their topological patterns of anatomical or functional connectivity [8].

Synchronization is an important collective motion in nature, which can be widely seen in neuroscience, computer science and telecommunication [13]–[20]. It has been demonstrated that synchronization of distributed brain activity plays an intriguing role in neural information processing [21]–[25]. In [21], it was revealed that inter-areal phase synchrony in the different frequency bands among frontoparietal and visual regions of a brain shows a systems level mechanism for coordinating and regulating the maintenance of neuronal object representations in visual working memory. Additionally, in [23], it was found experimentally that certain chaos in the brain, such as schizophrenia, epilepsy, autism, Alzheimer's disease, and Parkinson's disease, are closely related to abnormal neural synchronization.

Controllability of complex networks is an important problem in the area of physics and control theory, aiming at reducing the cost of regulating or coordinating the states of the network to a desired state. The driver nodes can also be referred as leaders or controlling areas, by injecting feedback information to some key nodes [26]–[32]. As demonstrated in [29], control the behaviors of a complex network is a common requirement in different kinds of networks, such as physical, transportation, and neuronal networks [21]. Up to now, various kinds of approaches have been developed to investigate the problem of controllability (pinning control) of complex networks. In [33], based on the results of [34], the problem of pinning controllability of complex networks was investigated in terms of the spectral properties of an extended network topology. Ref. [18] examined the distributed pinning synchronization of stochastic coupled neural networks via randomly occurring con-

---

- *Yang Tang is with the Institute of Physics, Humboldt University of Berlin, Berlin 12489, Germany and the Department of Transdisciplinary Concepts and Methods, Potsdam Institute for Climate Impact Research, Potsdam 14415, Germany. E-mail: tangtany@gmail.com.*
- *Huijun Gao is with the Research Institute of Intelligent Control and Systems, Harbin Institute of Technology, Harbin 150080, China, China. E-mail: hjgao@hit.edu.cn.*
- *Jürgen Kurths is with the Potsdam Institute for Climate Impact Research, Potsdam, Germany and Institute of Physics, Humboldt University of Berlin, Berlin, Germany and Institute for Complex Systems and Mathematical Biology, University of Aberdeen, Aberdeen AB24 3UE, United Kingdom. E-mail: Juergen.Kurths@pik-potsdam.de.*

Digital Object Indentifier 10.1109/TCBB.2013.72

trol. An optimization method to find the minimum number of pinned nodes was presented for ensuring a distributed mean square synchronization by utilizing semi-definite programming. In [35], the pinning impulsive controllers were presented to drive the whole state-coupled dynamical network to some desired trajectory. In [29], the controllability of an arbitrary complex directed network was investigated and the set of driver nodes was identified. When analyzing real networks, it was found that the number of driver nodes is determined mainly by the network's degree distribution. In [36], pinning control of undirected complex networks was investigated by some novel particle swarm optimization algorithms.

As shown in [21], [29], [37]–[39], it remains of great significance to study the controllability of a neuronal network, which can not only provide deep insights into understanding how to control real weighted and directed networks, but also help to understand how to avoid abnormal synchronization to inhibit neural diseases such as schizophrenia, epilepsy, autism, Alzheimer's disease, and Parkinson's disease [23], [40]. Based on single-objective evolutionary computation strategies, the identification of controlling regions was simply investigated in the neuronal network of cats' brain in microscopic, mesoscopic and macroscopic scales [9]. The controlling areas found in [9] are different from the usual hubs detected by degree, betweenness centrality, motif and synchronization [40], [41]. It was unveiled that the community Auditory in cats' brain, which is sparsely connected with other communities, is the most important community to control the entire neuronal network. Furthermore, in order to embrace two measures of controllability into one unified framework, the problem of identification of controlling regions in the neuronal network of cats' brain was investigated by means of an improved constrained optimization evolutionary algorithm [38], in which one important measure is regarded as an objective and the other measure is viewed as a constraint. By relaxing the constraint gradually, the controlling regions were identified. However, although the controlling regions were identified under different levels of constraints, one has to tune the value of constraint carefully. *A natural question arises here: is it possible to regard two measures of controllability of the neuronal network equally and identify the controlling regions under different levels of constraints at the same time?* Therefore, the first motivation of this paper is to shorten such a gap by contributing the first attempts to deal with multiobjective identification of control regions in the neuronal network, which can be formulated as a multiobjective optimization problem.

Evolutionary algorithms have been widely used in various optimization problems such as state estimation [42] and filtering [43]. In a multiobjective optimization problem, multiobjective optimization evolutionary algorithms (MOEAs) have attracted increasing attention, since they have the capability of dealing with a multiobjective optimization problem as well as finding nondominated sets in a single run [44]. Among them, nondominated sorting genetic algorithm-II (NSGA-II) [45] and strength Pareto evolutionary approach-2 (SPEA 2) [46] are the most popular ones. Recently, in order to lead a decision-maker to the most preferred solution of ones's choice, reference-point based methods have been proposed in MOEAs [47]. The progress towards the most preferred solution is interactive and reference-point based MOEAs have shown their effectiveness in real world applications when man is included in the optimization process. In addition, recent years have witnessed the increasing attention of differential evolution (DE), since DE is perceived as a reliable and versatile population-based heuristic optimization method [48]–[50]. Recently, a composite DE (CoDE), has been proposed to utilize three trial vector generation strategies and three control parameter settings [51]. It has been shown that CoDE delivers a promising performance in various single objective optimization problems. Although the performance of NSGA-II has become a standard method to deal with multiobjective optimization problems, there still exists some research room to improve the search performance of NSGA-II to enhance the controllability in a neuronal network by replacing the search engine using CoDE and including the idea of a reference-point based technique into NSGA-II, which is the second incentive of this paper.

Motivated by the above discussion, the problem of multiobjective identification of controlling areas in the neuronal network is investigated by means of a reference point based nondominated sorting composite differential evolution (RP-NSCDE). Solutions to the controllability of a neuronal network can be shown as pareto fronts (PFs) by using the proposed RP-NSCDE. In order to validate the effectiveness of RP-NSCDE, it is compared with statistical methods in complex network theory [1] and NSGA-II [45]. In addition, it is also compared with the results in [9] and [38], in which single objective evolutionary algorithms and constraint optimization evolutionary algorithms are utilized, respectively. The contributions of this paper can be summarized as follows: (1) the problem of multiobjective identification of controlling areas in a neuronal network is studied for the first time; (2) an improved MOEA, i. e., RP-NSCDE is proposed to show its reliability and effectiveness of multiobjective identification of controlling areas in a neuronal network; (3) the obtained results are compared with recent results in [9] and [38] to show the advantages of this work.

The organization of this paper is listed as follows. Sec. II presents the models and methods of the multiobjective identification of controlling areas in a neuronal network. In Sec. III, RP-NSCDE is presented in detail and the experiments are performed to validate the effectiveness of RP-NSCDE. Conclusion and discussion are given in Sec. IV.

Notations: In this paper, $l \in [1, N]$ represents the number of driver nodes of a network, where N is the network size. $\delta_{\mathcal{D}}(\cdot)$ denotes the characteristic function of the set $\mathcal{D}$, i.e., $\delta_{\mathcal{D}}(i) = 1$ if $i \in \mathcal{D}$; otherwise, $\delta_{\mathcal{D}}(i) = 0$. Define a graph by $\mathcal{G} = [\mathcal{V}, \mathcal{E}]$, where $\mathcal{V} = \{1, \cdots, N\}$ and $\mathcal{E} = \{e(i,j)\}$ are the vertex set and the edge set, respectively. The graph $\mathcal{G}$ is assumed to be directed, weighted and simple. Let the weighted and directed matrix $G = [g_{ij}]_{i,j=1}^{N}$ be the adjacency matrix of cat graph $\mathcal{G}$, which is defined as follows: for any pair $i \neq j, g_{ij} < 0$ if $e(i,j) \in \mathcal{E}$; otherwise, $g_{ij} = 0$. $g_{ii} = -\sum_{j=1, j\neq i}^{N} g_{ij}$ $(i = 1, 2, \cdots, N)$. The Laplacian matrix L is defined as follows: for any pair $i \neq j, l_{ij} = -1$ if $e(i,j) \in \mathcal{E}$; otherwise, $l_{ij} = 0$. $l_{ii} = -\sum_{j=1, j\neq i}^{N} l_{ij}$, $(i \in \mathcal{V})$. The output-degree $k_{out}(i) = -\sum_{j=1, i\neq j}^{N} l_{ij}$ of a node i is the number of efferent connections that it projects to other nodes, and its input-degree $k_{in}(i) = -\sum_{j=1, i\neq j}^{N} l_{ji}$, is the number of the afferent connections it receives.

2 MODELS AND METHODS

2.1 The graph of cats' brain

The neuronal network of cats' brain, which shows the anatomical connectivity [52]–[56], is a typical weighted and directed complex network. The detailed connection matrix and analysis of neuronal networks of cats' brain was given in [52], which is obtained from several subtle steps including cortical parcellation, thalamic parcellation, collation of connection data and translation from database to connection matrix. For more details regarding the construction of the connection matrix, please refer to [52] and references therein. The neuronal network can be separated into 53 cortical regions ($N = 53$) with about 850 fibres of different densities. This neuronal network has been found to exhibit short average pathlength and high clustering coefficient, indicating an optimal coordination for effective inter-area communication and for achieving high functional complexity [41]. The cat cortical network also exhibits a hierarchically clustered organization [53], where it is composed of four topological clusters coinciding with four functional cortical communities: visual cortex (16 areas), auditory (7 areas), somato-motor (16 areas) and fronto-limbic (14 areas).

2.2 Multiobjective controllability of neuronal networks

Let a reference evolution (desired state) be written as follows:

$$\frac{ds(t)}{dt} = f(s(t)). \tag{1}$$

As mentioned in [37], this differential equation is general enough to represent wide real-world complex systems such as biological networks, transportation networks and other natural systems.

Based on (1), the neuronal network with identical subsystems and a feedback controller can be written as:

$$\begin{aligned}\frac{dx_i(t)}{dt} &= f(x_i,t) - a\sum_{j=1}^{N} g_{ij}h(x_j(t)) \\ &\quad - a\delta_{\mathcal{D}}(i)v_i(h(s(t)) - h(x_i(t))), i \in \mathcal{V}, \end{aligned} \tag{2}$$

where $x_i(t) = [x_{i1}(t), x_{i2}(t), \cdots, x_{in}(t)]^T \in \mathbb{R}^n$ $(i \in \mathcal{V})$ is the state vector of the ith area and $f(x_i,t) = [f_1(x_i,t), \cdots, f_n(x_i,t)]^T$ is a continuous vector function; n denotes the dimensional size of each area; a is the global coupling gain of the network; $h(x_i(t))$ is the output function; G is the coupling matrix of neuronal network of cats' brain. Let $\mu_p = \mu_p^r + j\mu_p^m (j = \sqrt{-1}), (p \in \mathcal{V})$, be the set of eigenvalues of C and suppose that they are ordered in such a way that $\mu_1^r \leq \mu_2^r \leq \cdots \leq \mu_N^r$. $v_i (i \in \mathcal{V})$ is the control gain or coupling strength between the area and the desired state. It is clear that $1 \leq \sum_{i=1}^{N} \delta_{\mathcal{D}}(i) \leq N$. The purpose of pinning control is to drive the state of the neuronal network in (2) toward the desired state $s(t)$ in (1), i. e., $x_1(t) = x_2(t) = \cdots = x_N(t) = s(t)$.

In order to measure the controllability of the neuronal network (2), an extended network of $N+1$ dynamical systems y_i can be formulated, where $y_i = x_i$ for $i \in \mathcal{V}$ and $y_{N+1} = s$ as follows [33], [38]:

$$\begin{aligned}\frac{dy_i(t)}{dt} &= f(y_i,t) - a\sum_{j=1}^{N+1} \mathcal{R}_{ij}h(y_j(t)), \\ &\qquad i \in \mathcal{V} \cup \{N+1\}, \end{aligned} \tag{3}$$

where $\mathcal{N} = [\mathcal{R}_{ij}] \in \mathbb{R}^{(N+1)\times(N+1)}$ in the form of

$$\mathcal{N} = \begin{pmatrix} \mathcal{S}_1 & g_{12} & \ldots & g_{1N} & -\delta_{\mathcal{D}}(1)v_1 \\ g_{21} & \mathcal{S}_2 & \ldots & g_{2N} & -\delta_{\mathcal{D}}(2)v_2 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ g_{N1} & g_{N2} & \ldots & \mathcal{S}_N & -\delta_{\mathcal{D}}(N)v_N \\ 0 & 0 & \ldots & 0 & 0 \end{pmatrix}, \tag{4}$$

in which $\mathcal{S}_i = g_{ii} + \delta_{\mathcal{D}}(i)v_i$. Let $\lambda_p = \lambda_p^r + j\lambda_p^m$ denote the pth eigenvalue of $\mathcal{N}$ and suppose that λ_p is sorted as a sequence $\lambda_1^r \leq \lambda_2^r \leq \cdots \leq \lambda_{N+1}^r$, where $\lambda_1^r = 0$.

According to the analysis method of checking synchronizability of complex networks [33], [34], the controllability can be assessed in terms of minimizing the following two objectives:

$$f_1 = \min \frac{\lambda_{N+1}^r}{\lambda_2^r}, \tag{5}$$

and

$$f_2 = \min \max_{p} \{\lambda_p^m\}. \tag{6}$$

As discussed in [33], [34], for objective (5), to satisfy synchronization, all the $a\lambda_i^r, i = 2, ..., N+1$, should belong to the bounded region of the complex plane, where the master stability function is negative. For objective (6), the condition on the maximum imaginary part of the spectrum provides information about the distribution of the eigenvalues along the direction of the imaginary axis.

Note that both [29] and this paper investigate the problem of controllability of networks. However, the main differences between both are listed as follows: 1) this paper treats controllability of networks as a multiobjective optimization problem and thus a multiobjective optimization evolutionary algorithm is utilized to analyze controllability, whereas [29] utilized controllability in the control theory to handle pinning control of networks; 2) the purpose of [29] is to identify the minimum number of nodes to control the entire networks, while the proposed method in this paper can also investigate controllability under different numbers of driver nodes l.

Remark 1. Recently, controllability of complex networks has attracted increasing attention. In [33], the authors defined two measures for controllability of complex networks based on the synchronizability of them [34]. The authors utilized the information of degree to choose driver nodes and tune their control gains by simply assuming the control gain in each node is equal to each other. In order to improve the conservativeness of the results in [28], [33], single objective evolutionary algorithms were proposed to select the driver nodes and control gains. It is shown that the methods can enhance the controllability of complex networks [36]. Single objective evolutionary algorithms were then applied to identify the controlling areas in the neuronal network by minimizing f_1 and f_2 separately [9], which inevitably neglects the nature of a directed and weighted topology of the neuronal network. In order to include f_1 and f_2 into one unified framework, the problem of identification controlling areas was transformed into a constraint optimization problem, in which f_1 is regarded as an objective and f_2 is considered as a constraint [38]. Then, an improved constraint optimization evolutionary algorithm was developed to identify the controlling regions in the neuronal network under different levels of f_2. However, although the results in [38] can solve the problem of detecting controlling regions in the neuronal network, one has to adjust f_2 blindly, which will lead to some inefficiency to show all the solutions under different values of f_2 in a single run. In other words, the methods in [38] fail to show PFs in a single run. Different from the results in [9] and [38], in this paper, we formulate the controllability of the neuronal network into a multiobjective optimization problem with two conflicting objectives f_1 and f_2. In the following, an improved MOEA is developed and is used to investigate the multiobjective identification of controlling areas of the neuronal network.

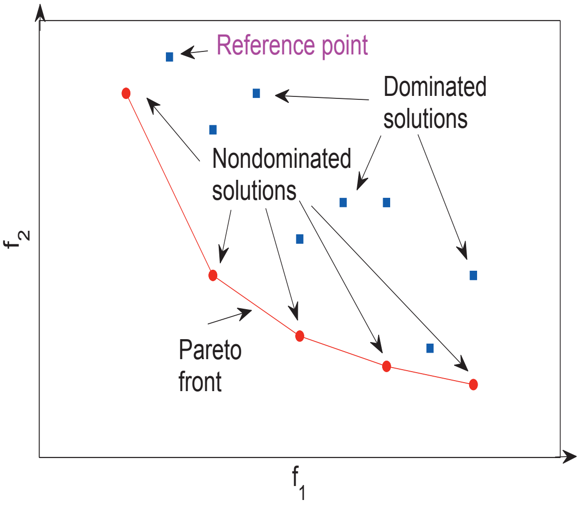


Fig. 1. Illustration of the optimal Pareto front, the reference point, the relationship between dominated and nondominated solutions.

2.3 Methods

In this subsection, the concept of multiobjective optimization and an improved MOEA are presented. The improved MOEA includes a nondominated sorting mechanism [45], a diversity preservation mechanism [45], the replacement of genetic algorithms with differential evolution and a reference point approach. The detailed procedure is presented here.

2.3.1 Concepts of Multiobjective Optimization

Without any loss of generality, a minimization problem is investigated with a decision space Ω, where f_1 in (5) and f_2 in (6) are two objectives regarding controllability of the neuronal network and should be minimized as small as possible simultaneously. We aim to solve multiobjective optimization problems for a variable set $\mathcal{W}$ that optimizes the following objective:

$$\min_{\mathcal{W}\in\Omega} F(\mathcal{W}), \mathcal{W} \in \mathbb{R}^D, \tag{7}$$

where D is the dimension size of an optimization problem, $\mathcal{W} = \{w_1, w_2, \cdots, w_D\}$ is a vector with a set of decision variables and $F = \{f_1, f_2, \cdots, f_m\}$ is the objective vector with m objectives to be minimized. In this paper, $m = 2$ and $f_i(i = 1, 2)$ is given in (5) and (6), respectively.

The following definitions of Pareto dominance and Pareto optimality are useful for presenting our results and are fundamental in multiobjective optimization, with Pareto dominance forming the basis of the equality of the solution.

Definition 1. *(Pareto Dominance): Given the objective vectors* $Y_1 \in \mathbb{R}^m$ *and* $Y_2 \in \mathbb{R}^m$*, then* Y_1 *dominates* Y_2*, denoted as* $Y_1 \prec Y_2$*, iff* $y_{1i} \leq y_{2i}, \forall i \in \{1, 2, \cdots, m\}$ *and* $y_{1i} < y_{2i}, \exists i \in \{1, 2, \cdots, m\}$.

Definition 2. *(Optimal Pareto Front): The optimal Pareto front (PF) denoted by $\mathcal{F}^*$ is the set of individuals*

$$\mathcal{F}^* = \{\mathcal{F}_j^* | \mathcal{F}_j^* \prec \mathcal{F}_i, \forall \mathcal{F}_i \in \mathcal{F}\}. \tag{8}$$

It is worth mentioning that, different from the single objective optimization problem [9], there exists a set of solutions to the multiobjective optimization problem, which can be shown in the form of PFs. Our first purpose is to develop efficient multiobjective evolutionary algorithms to tackle the problem of multiobjective controllability of the neuronal network, which would find PFs with both a high accuracy and a fast convergence speed.

2.3.2 NSGA-II

NSGA-II was developed in [45], which is composed of a fast nondominated sorting approach and a diversity preservation mechanism. NSGA-II reduces the computational complexity and includes an elitism approach to prevent the loss of good solutions. In addition, NSGA-II is adaptive and does not require to define a specific sharing parameter. It is well recognized that NSGA-II is efficient to handle multiobjective optimization problems in both theoretical and practical aspects.

Remark 2. It is worth mentioning that the concept of multiobjective optimization has been used in considering the problem of multiobjective synchronization of two linearly coupled systems [57], in which control cost and convergence speed are minimized simultaneously. The constraints on the coupling form are also concerned by converting the multiobjective synchronization into a multiobjective constraint problem. By utilizing an improved NSGA-II, it is found that there exist PFs when optimizing two performance measures.

2.3.3 Composite DE (CoDE)

CoDE was proposed in [51], which aims to deal with single objective optimization and shows its effectiveness in unimodal functions, multimodal functions, expanded multimodal functions and hybrid composition functions. The primary idea of CoDE is to randomly combine several trial vector generation strategies with a number of control parameter settings at each generation to create new trial vectors. CoDE is composed of three trial vector generation strategies and three control parameter settings. The three trial vector generation strategies are the following:

1) "rand/1/bin":

$$u_{i,j,G} = \begin{cases} x_{r1,j,G} + F * (x_{r2,j,G} - x_{r3,j,G}), \\ \text{if } rand \le C_r \text{ or } j = j_{rand} \\ x_{i,j,G}, \\ \text{otherwise.} \end{cases} \tag{9}$$

2) "rand/2/bin":

$$u_{i,j,G} = \begin{cases} x_{r1,j,G} + F * (x_{r2,j,G} - x_{r3,j,G}) \\ +F * (x_{r4,j,G} - x_{r5,j,G}), \\ \text{if } rand \le C_r \text{ or } j = j_{rand}, \\ x_{i,j,G}, \\ \text{otherwise.} \end{cases} \tag{10}$$

3) "current-to-rand/1":

$$\begin{aligned} u_{i,j,G} &= x_{i,j,G} + rand * (x_{r1,j,G} - x_{i,j,G}) \\ &\quad + F * (x_{r2,j,G} - x_{r3,j,G}). \end{aligned} \tag{11}$$

where $i = 1, 2, ..., SP$ and SP is the population size; $j = 1, 2, ..., D$ and D is the dimension size; G is the generation number; $x_{i,j,G}$ is the state of ith individual in jth dimension at generation G; $r1$, $r2$, $r3$, $r4$ and $r5$ distinct random integers from the interval $[1, SP]$ and are also different from i. F is the scaling factor, which amplifies the difference vectors; C_r is the crossover control parameter; j_{rand} is an integer from the interval $[1, D]$; $rand$ is a uniformly distributed random number between 0 and 1.

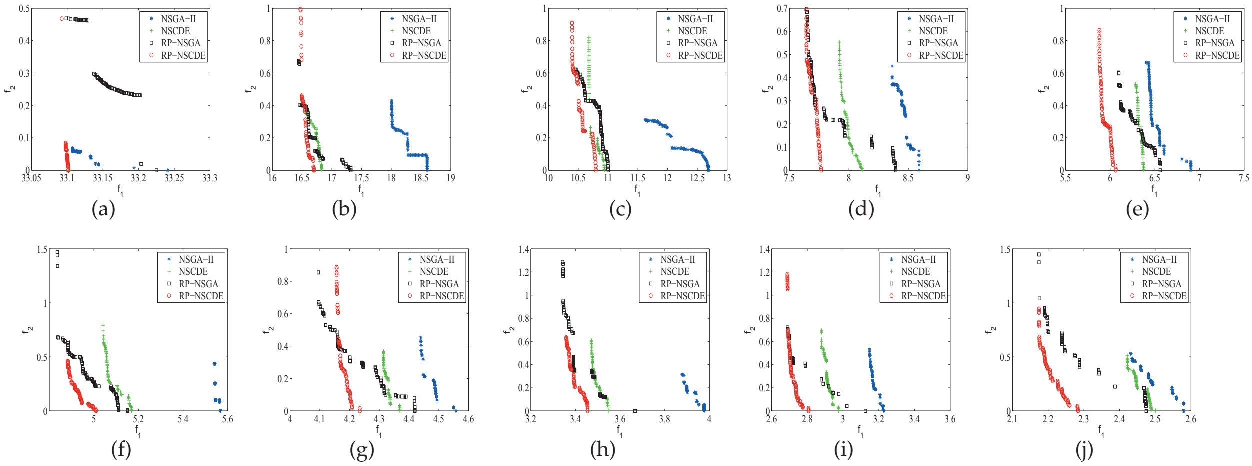
The three control parameter settings are:

1) $F = 1, C_r = 0.1$;
2) $F = 1, C_r = 0.9$;
3) $F = 0.8, C_r = 0.2$.

The first control parameter setting, $[F = 1.0, C_r = 0.1]$, is for dealing with separable problems; the second control parameter setting, $[F = 1.0, C_r = 0.9]$, is aimed to maintain the population diversity and thus enhances the ability of global exploration; and the last control parameter setting, $[F = 0.8, C_r = 0.2]$, is for the exploitation of the three strategies and thus increases the convergence speed of the population [51]. At each generation, each trial vector generation strategy in the strategy candidate pool (9)-(11) is used to create a new trial vector with a control parameter setting randomly chosen from three parameter settings. Thus, three trial vectors are generated for each target vector and the best one is chosen to enter the next generation if it is better than its target vector.

2.3.4 Adaptive differential evolution (JaDE)

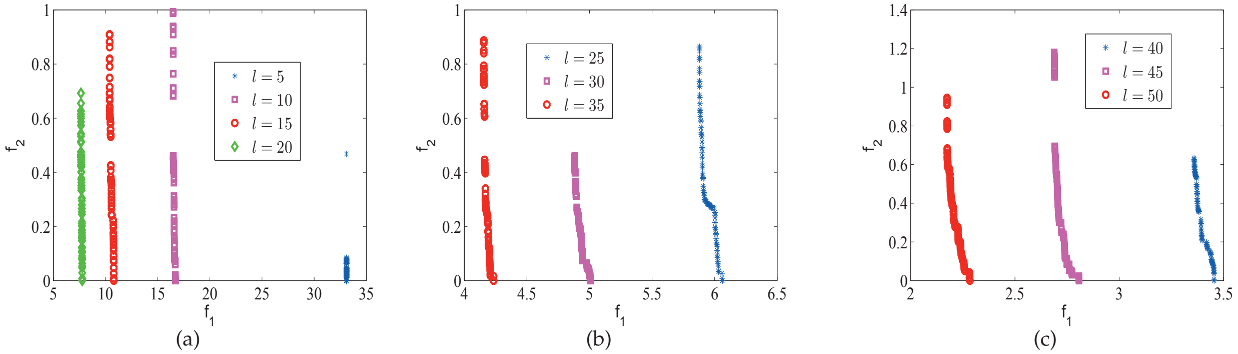
JaDE is made up of an adaptive updated mechanism with a Cauchy distribution and a Normal distribution [48]. The DE/current-to-ϵbest strategy is intoduced to maintain multiple best solutions to balance the convergence speed and the diversity of the population. In the updating equation, the current population information and an external archive are used to generate new individuals. The archive is employed to store the recently explored inferior solutions. In addition, each individual has its own scaling factor and crossover control parameter, which are generated according to a Normal distribution and a Cauchy distribution. JaDE has been demonstrated its advantage and effectiveness in various single objective optimization problems, such as unimodal optimization



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 2. Pareto fronts obtained by NSGA-II, NSCDE, RP-NSGA and RP-NSCDE under different l. (a) PFs under $l = 5$; (b) PFs under $l = 10$; (c) PFs under $l = 15$; (d) PFs under $l = 20$; (e) PFs under $l = 25$; (f) PFs under $l = 30$; (g) PFs under $l = 35$; (h) PFs under $l = 40$; (i) PFs under $l = 45$; (j) PFs under $l = 50$.



(a) (b) (c)

Fig. 3. The comparison of PFs by RP-NSCDE under different l. (a) PFs obtained by RP-NSCDE with $l = 5$, $l = 10$, $l = 15$ and $l = 20$; (b) PFs obtained by RP-NSCDE with $l = 25$, $l = 30$ and $l = 35$; (c) PFs obtained by RP-NSCDE with $l = 40$, $l = 45$ and $l = 50$.

problems, multimodal optimization problems, rotated optimization problems, non-continuous optimization problems and hybrid optimization problems, etc [48]. JaDE has the ability of achieving a high accuracy as well as a fast convergence speed. In [9], it is also shown that JaDE outperforms other well-known single objective optimization methods in dealing with the controllability of the neuronal network, in which f_1 and f_2 are optimized separately. For more details regarding JaDE, please refer to [48].

2.3.5 A reference point based nondominated sorting composite differential evolution (RP-NSCDE)

NSGA-II has been verified its promising performance in various multiobjective problems and real world applications [45], [57]. Nevertheless, the search engine in NSGA-II is genetic algorithms with simulated binary crossover (SBX) operator and polynomial mutation, which might have limitations in exploration and exploitation in the search space. In addition, NSGA-II does not use the useful information to explore the preferred region and thus explores the search space blindly. Therefore, there still remains much research room to develop more efficient MOEAs by replacing inefficient search engines-genetic algorithms, while maintaining the efficient parts of NSGA-II, i. e., the nondominated sorting approach and the diversity preservation mechanism.

Here, we develop a reference point based nondominated sorting composite differential evolution (RP-NSCDE). Due to the satisfactory performance of CoDE in dealing with single objective optimization and its simple mechanism, CoDE can be easily applied to multiobjective optimization problems without complicated additional modifications. In addition, we utilize a more “intelligent” mechanism: by introducing a reference point into the MOEA. The reference point is usually provided by users to save the computation time and guide the population to search more desirable areas of users [47], [58]. Since the capability of JaDE was shown in the identification of controlling regions in the neuronal network with a quick conver-

Algorithm 1 A reference point based nondominated sorting composite differential evolution (RP-NSCDE)

```
Begin
    Create a random population $P_t (t = 0)$ in feasible solution space with $SP - 1$ individuals.
    while $f_e \leq f_{e,\max}$ do
        $\tilde{P}$ = reference-point-generation($\mu * f_{e,\max}$); /*generating a reference point $\tilde{P}$ by minimizing $f_1$ with a fitness evaluation $\mu * f_{e,\max}$, where $\mu$ is predefined value equaling to $0.2$.*/;
        $P_t$ = add-into-population($P_t, \tilde{P}$); /*add the reference point into the population $P_t$ and the population of $P_t$ is $SP$.*/;
        $M_t$ = tournament-selection($P_t$); /*using a binary tournament selection to generate a parent population with $\lfloor SP/2 \rfloor$ individuals*/;
        $Q_t$ = composite-differential-evolution($M_t$); /*offspring $Q_t$ is generated according to (9)-(11) and its population size of $Q_t$ is 3*$\lfloor SP/2 \rfloor$*/;
        $S_t = M_t \cup Q_t$; /*combine parent and offspring population*/;
        $\mathcal{F}$ = nondominated-sort($S_t$); /*$\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2, ...)$ contains all nondominated fronts of $S_t$*/;
        $P_{t+1} = \varnothing$ and $i = 1$
        while $|P_{t+1}| + |\mathcal{F}_i| \leq SP$ do /*$|.|$ is the cardinality of a set*/;
            crowding-distance-assignment($\mathcal{F}_i$); /*calculating crowding distance of $\mathcal{F}_i$*/;
            $P_{t+1} = P_{t+1} \cup \mathcal{F}_i$; /*include $i$th nondominated front in the population*/;
            $i = i + 1$;
            sort($\mathcal{F}_i$, $\prec$); /*sort the individuals in $\mathcal{F}_i$ using $\prec$*/;
            $P_{t+1} = P_{t+1} \cup \mathcal{F}_i[1 : (NP - |P_{t+1}|)]$; /*fill $P_{t+1}$ with the best $(SP - |P_{t+1}|)$ individuals in $\mathcal{F}_i$*/;
            $t = t + 1$.
        end while
    end while
End
```

gence speed [9], JaDE is used to provide the reference point by minimizing f_1 with a predefined number of fitness evaluation. It is worth mentioning that we do not utilize JaDE to minimize f_2 in RP-NSCDE, since only optimizing f_2 will neglect to optimize f_1, as seen in Table 3.

Before presenting the procedure of RP-NSCDE, we give some preliminaries first. f_e stands for the fitness evaluation and $f_{e,\max}$ is the maximum number of fitness evaluation allowed. Let $\prec$ be a partial order, representing that between two solutions with different nondomination ranks (belonging to different fronts), the solution with the lower (better) rank is preferred. Otherwise, if both solutions belong to the same front, the solution located in a less crowded region is preferred. The reference-point-generation(.) function is to utilize JaDE to provide a reference point by only optimizing f_1 when f_e achieves $\mu * f_{e,\max}$. The add-into-population(.,.) is to add the reference point to the population and make the population has SP individuals. μ is a predefined constant to balance the trade-off between obtaining a more accurate reference point (more close to the optimal PF) and spreading the solutions in PFs uniformly. Here, $\mu = 0.2$ is adopted, since the observation from [9] shows that the solutions achieved by JaDE usually converge with such a value of fitness evaluation. The nondominated-sort(.) function is to generate different levels of PFs according to pareto dominance, which was given in [45]. In addition, the diversity preservation mechanism is to make the solutions widely spread in the obtained set of solutions [45]. Based on these mechanisms, the pseudocode of RP-NSCDE is given in Algorithm 1.

3 EXPERIMENTS

3.1 Experimental information

The encoding scheme follows from [9], [38], which consists of two components with the equal dimension size l: the first part is an integer search space to denote the locations of driver nodes; the other part is a continuous search space to stand for the control gains of driver nodes. Therefore, the dimension size is $D = 2 * l$. For a detailed example, please refer to [9], [38]. In addition, the population size of NSGA-II and RP-NSCDE is $SP = 100$. The maximum number of fitness evaluation allowed $f_{e,\max} = SP * D * \kappa$ is for all the evolutionary algorithms in this paper, where SP is the population size, D is the dimension size and κ is a predefined parameter. It is worth mentioning that κ is an adjustable parameter to get a balance between complexity and search accuracy. Usually, a large κ is helpful to enhance search performance but leads to huge computation complexity. A small κ can save the computation resources but may result in unsatisfactory search performance. In this paper, we set $\kappa = 175$ to make a balance between complexity and search accuracy. The parameter settings for JaDE and constraint optimization methods follow from [9], [38], [48], [59]. Each evolutionary algorithm is carried out for 20 runs for eliminating discrepancy.

3.2 Methods for comparison

In this subsection, some methods are given for comparisons with RP-NSCDE to show the effectiveness of RP-NSCDE. The methods include statistical methods from complex network theory [1], [9], a single objective optimization method [9], [48], constraint optimization methods [38], [59] and multiobjective optimization methods. JaDE and NSGA-II are given in Sec. 2.3.4 and Sec. 2.3.2, respectively and thus they are omitted here.

3.2.1 Statistical strategies in complex network theory

In order to characterize the properties of complex networks, it is well known that degree, between-centrality and closeness are popular measures in complex network theory [1]. In [41], the information of degree, between-centrality and closeness were used to detect the hubs in the neuronal network of cats' brain. In controllability of complex networks, such information of complex networks has been used to find controlling regions of complex networks or neuronal networks [9], [60]. Here, following the work in [9], we can select the driver nodes in the neuronal network of cats' brain by sorting degree, between-centrality (BC) and closeness of each node ascendingly (descendingly) in the neuronal network, respectively. We summarize the statistical strategies in complex network theory in Table 1. In statistical strategies, control gains are assumed to be identical in each driver node and then tuned gradually from an interval $[0, N]$ with a step size 0.01.

TABLE 1
Statistical strategies in complex network theory and their abbreviations.

Methods	Abbreviation
Ascending degree-based strategy	ADBS
Descending degree-based strategy	DDBS
Ascending betweenness centrality-based strategy	ABCBS
Descending betweenness centrality-based strategy	DBCBS
Ascending closeness-based strategy	ACBS
Descending closeness-based strategy	DCBS

3.2.2 Constraint optimization methods-a dynamic hybrid framework (DyHF) and an improved dynamic hybrid framework (IDyHF)

DyHF was very recently proposed in [59], which includes both a global search and a local search. In both global and local search models, differential evolution (DE) works as a search algorithm, and Pareto dominance in multiobjective optimization is employed to deal with constraint handling. DyHF formulates a constrained optimization problem into a biobjective optimization problem by regarding the degree of constraint violation as an additional objective. The global search model mainly concentrates on refining the overall performance of the population and exploring more promising regions. The local search model aims to guide the population to feasible regions quickly by avoiding the case of stagnating in infeasible regions. In the local search model, the population can be guided to feasible regions from different directions.

Although DyHF has been shown its effectiveness and performance in constraint optimization problems by carrying out a scientific set of experiments [59], the DE in global search model is not adaptive and thus DyHF does not have the ability of fitting the circumstances adaptively. In order to make the DyHF more powerful to have a high accuracy, an efficient constraint handling and a quick search speed, a self-adaptive differential evolution named jDE [49] was used to replace the search engine in DyHF and therefore an IDyHF was proposed in [38]. It was found that IDyHF can efficiently enhance the performance of DyHF for the identification of controlling regions in the neuronal network.

3.3 Comparison

In this subsection, in order to show the advantages of the proposed RP-NSCDE, we compare it with different methods in complex network theory, single objective optimization problems, constraint optimization problems and multiobjective optimization problems, as given in Sec. 3.2.

3.3.1 Comparison of statistical strategies in complex network theory and RP-NSCDE

The comparison results of statistical strategies in complex network theory and RP-NSCDE are provided in Table 2 by optimizing f_1 and f_2 separately. Note that RP-NSCDE is a multiobjective optimization method and thus has a set of nondominated solutions. Therefore, we select the solution with the minimum f_1 among a set of nondominated solutions and record its corresponding f_2 in each run. Then, after 20 runs, the mean value of f_1 and f_2 can be calculated. When minimizing f_1, it can be seen that RP-NSCDE achieves best for f_1 in all the cases and the corresponding values of f_2 achieve best in six cases. It is worth mentioning that ADBS, ABCBS and DCBS usually perform better than DDBS, DBCBS and ACBS in optimizing f_1, which implies that driver nodes are usually selected from areas with a small degree. Therefore, driver nodes from a small degree will deliver a robust performance in minimizing f_1. As l increases, f_1 becomes smaller in all the methods. That is, the more driver nodes are chosen, the better the controllability of the neuronal network is.

When minimizing f_2, RP-NSCDE works best among the seven methods in both f_1 and f_2 under different l. It can be seen that RP-NSCDE can not only maintain f_2 to attain zero but also deliver the best value in f_1 among the seven methods. For RP-NSCDE, as l monotonically increases, f_1 decreases gradually and maintains f_2 at 0. All the methods in complex network theory cannot suppress f_2 to zero successfully. Different from RP-NSCDE and the case of minimizing f_1, f_2 does not decrease for all the methods from complex network theory, as l increases. Like the case of minimizing f_1, the regions with a small degree are still robust to minimize f_2 and their corresponding f_1. To summarize, RP-NSCDE performs best among these methods in both minimizing f_1 and f_2 under a different allowed number of driver nodes.

TABLE 2

Comparison of statistical strategies in complex network theory and RP-NSCDE for the identification of controlling regions of the neuronal network of cats' brain when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50. The mean results of minimum of f_1 (f_2) and its corresponding f_2 (f_1) among the nondominated solutions in 20 runs are provided for RP-NSCDE. The better results between statistical strategies and RP-NSCDE are shown in **Bold fonts**.

	Minimization f_1													
	DDBS		DBCBS		ACBS		ADBS		ABCBS		DCBS		RP-NSCDE	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
$l = 5$	71.4213	0.9111	55.9645	0.9363	71.4213	0.9111	56.4496	1.0737	43.0772	0.8252	65.2997	0.8439	**33.1132**	**0.2813**
$l = 10$	44.1461	0.8214	36.9477	0.7972	44.1461	0.8214	24.3185	0.8698	22.6184	0.9679	29.0945	0.9823	**16.5113**	**0.7962**
$l = 15$	34.2414	0.7816	31.7341	0.8994	34.2414	0.7816	14.1849	0.9779	14.0409	1.3173	15.0466	0.8193	**10.7217**	**0.7012**
$l = 20$	32.4692	0.7786	22.7841	**0.3584**	31.9017	0.7581	9.4899	1.2315	11.5303	1.0777	10.1022	1.0053	**7.8189**	0.4374
$l = 25$	29.3212	0.3492	17.9532	0.5004	29.7578	**0.3372**	7.4427	1.3211	9.6757	1.0257	7.4714	0.9281	**6.166**	0.5735
$l = 30$	27.3839	0.3581	17.4414	0.3337	27.8319	**0.3428**	5.7293	1.4248	7.816	1.253	6.0127	1.7641	**5.0506**	0.4894
$l = 35$	26.0809	1.0203	17.1048	**0.3829**	25.9419	0.853	4.7137	1.282	6.1687	1.4648	4.7137	1.282	**4.1999**	0.711
$l = 40$	21.5184	1.0312	16.4416	0.9389	16.4793	0.9356	4.2255	1.543	4.4968	1.5531	4.4516	1.598	**3.4044**	**0.7588**
$l = 45$	16.3439	0.9287	15.3385	0.9168	15.6126	0.9058	3.2984	1.0562	4.0001	1.3338	3.2984	1.0562	**2.7106**	**0.8654**
$l = 50$	15.0445	0.8826	14.0634	0.8976	14.5706	0.8857	2.7175	0.8986	3.3332	0.9333	2.5691	**0.7928**	**2.1906**	0.9101
	Minimization f_2													
$l = 5$	809.9556	0.8849	615.738	0.8883	809.9556	0.8849	64.6466	0.3458	47.0724	0.8195	72.8859	0.6473	**33.5641**	**0**
$l = 10$	45.6937	0.7988	37.9129	0.7761	45.6937	0.7988	27.7794	0.7677	29.0126	0.6077	29.446	0.5544	**17.1039**	**0**
$l = 15$	35.1968	0.7641	37.6757	0.8565	35.1968	0.7641	16.2786	0.6296	37.022	0.5857	15.7334	0.5827	**11.0937**	**0**
$l = 20$	33.1039	0.7692	26.0629	0.3554	51.2308	0.5894	23.7415	0.5351	12.457	0.4681	13.3333	0.4471	**8.0189**	**0**
$l = 25$	57.1181	0.3398	21.8953	0.2171	30.934	0.3352	43.2995	0.464	11.0904	0.6343	8.8753	0.466	**6.2865**	**0**
$l = 30$	28.2536	0.3313	17.5127	0.3264	36.7119	0.3248	12.1871	0.4994	16.6001	0.5853	6.4633	0.3433	**5.1784**	**0**
$l = 35$	108.4975	0.9107	18.5507	0.3617	26.4119	0.8471	21.9108	0.7745	29.4098	0.529	21.9108	0.7745	**4.3846**	**0**
$l = 40$	89.1616	0.9196	43.8192	0.875	85.3365	0.8893	13.6795	0.7556	10.135	0.5872	17.7563	0.832	**3.5451**	**0**
$l = 45$	73.686	0.8916	72.0502	0.8952	72.662	0.8926	3.6277	0.5555	6.7121	0.6898	3.6277	0.5555	**2.8948**	**0**
$l = 50$	22.05	0.8775	60.1543	0.8922	62.6352	0.8838	3.0419	0.796	4.1913	0.6916	2.5855	0.7383	**2.5016**	**0**

TABLE 3

Comparison of JaDE and RP-NSCDE for the identification of controlling regions of the neuronal network when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50. The minimum results of optimization of f_1 (f_2) and its corresponding f_2 (f_1) in 20 runs are listed under different l. The better results between JaDE and RP-NSCDE are shown in **Bold fonts**.

	Minimization f_1				Minimization f_2			
	JaDE		RP-NSCDE		JaDE		RP-NSCDE	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
$l = 5$	33.097	0.4687	**33.0924**	**0.4677**	149.6125	0	**33.1018**	0
$l = 10$	16.4934	**0.9684**	**16.4717**	0.9936	46.014	0	**16.7009**	0
$l = 15$	10.4281	**0.7461**	**10.3936**	0.909	26.8588	0	**10.7954**	0
$l = 20$	**7.581**	0.7006	7.6431	**0.6925**	27.2322	0	**7.7681**	0
$l = 25$	**5.8517**	0.9622	5.8792	**0.8629**	52.4813	0	**6.0615**	0
$l = 30$	**4.6952**	0.9682	4.8831	**0.4617**	27.9384	0	**5.0096**	0
$l = 35$	**3.9439**	1.3907	4.1564	**0.8869**	125.8241	0	**4.2346**	0
$l = 40$	**3.2745**	1.1931	3.3585	**0.6344**	61.2819	0	**3.4546**	0
$l = 45$	**2.6768**	**0.9788**	2.6884	1.1802	63.4997	0	**2.8071**	0
$l = 50$	2.2029	1.7323	**2.1743**	**0.9449**	29.0712	0	**2.2841**	0

3.3.2 Comparison of JaDE and RP-NSCDE

In [9], the results of JaDE are given to identify controlling regions by optimizing f_1 and f_2 separately. We list the minimum values of f_1 (f_2) and their corresponding f_2 (f_1) in 20 runs under different l in Table 3. Note that RP-NSCDE is a multiobjective optimization method. Consequently, the minimum values of f_1 among the nondominated solutions and their corresponding f_2 are listed in Table 3 for RP-NSCDE in 20 runs under different l. Similarly, the results of minimizing f_2 are listed in Table 3. Although JaDE is a single objective optimization method and only focuses on minimizing one objective, JaDE cannot achieve better in all the cases than RP-NSCDE when minimizing f_1. RP-NSCDE performs better for the values of f_1 than JaDE under $l = 5, 10, 15$ and 50 when minimizing f_1. Although RP-NSCDE is worse than JaDE in other six cases, the difference between them is very small. All these observations demonstrate the effectiveness of RP-NSCDE, even if it is compared with a powerful single objective optimization algorithm-JaDE.

When minimizing f_2, both JaDE and RP-NSCDE can attain zero in all the cases, which are different from the statistical methods in complex network theory when minimizing f_2. However, it is obvious that all the nondominated solutions generated by RP-NSCDE dominate the solutions obtained by JaDE. It can be revealed that RP-NSCDE can offer much smaller f_1 than JaDE as well as maintaining f_2 at zero under different l, since RP-NSCDE is a multiobjective

TABLE 4
Comparison of DyHF, IDyHF and RP-NSCDE for the identification of controlling regions of the neuronal network when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50. The mean and minimum values of f_1 (f_2) and their corresponding f_2 (f_1) in 20 runs are given here. The best results among DyHF, IDyHF and RP-NSCDE are shown in **Bold fonts**.

$\delta = 0.1$												
	DyHF				IDyHF				RP-NSCDE			
	f_1		$\tilde{f}_2$		f_1		$\tilde{f}_2$		f_1		$\tilde{f}_2$	
	min	mean	min	mean	min	mean	min	mean	min	mean	min	mean
$l = 5$	33.1069	33.6454	0	0	33.1069	36.062	0	0	**33.0978**	**33.3285**	0	0
$l = 10$	17.6765	19.5482	0	0	17.4138	18.2775	0	0	**16.611**	**16.8277**	0	0
$l = 15$	12.566	13.4507	0	0	11.8687	12.4638	0	0	**10.7765**	**11.0554**	0	0
$l = 20$	10.1962	12.9785	0	0	8.5238	9.2097	0	0	**7.7525**	**7.9724**	0	0
$l = 25$	7.8375	9.988	0	0	6.9952	7.6494	0	0	**6.0223**	**6.2534**	0	0
$l = 30$	8.9547	10.0877	0	0	5.8847	6.4759	0	0	**4.944**	**5.1154**	0	0
$l = 35$	5.8531	9.1597	0	0	4.9418	5.3835	0	0	**4.2019**	**4.2846**	0	0
$l = 40$	5.9439	7.8927	0	0	3.8327	4.5113	0	0	**3.4441**	**3.4932**	0	0
$l = 45$	4.306	6.02	0	0	3.1575	3.8624	0	0	**2.7487**	**2.8342**	0	0
$l = 50$	4.0159	6.5301	0	0	2.4946	3.0037	0	0	**2.255**	**2.3936**	0	0
$\delta = 0$												
$l = 5$	38.1923	40.7439	0	0	33.1591	39.954	0	0	**33.1018**	**33.5641**	0	0
$l = 10$	18.8906	20.3117	0	0	18.2647	19.0401	0	0	**16.7009**	**17.1039**	0	0
$l = 15$	12.716	16.4437	0	0	12.2507	12.7915	0	0	**10.7954**	**11.0937**	0	0
$l = 20$	10.7616	12.9512	0	0	9.1151	9.5792	0	0	**7.7681**	**8.0189**	0	0
$l = 25$	9.9345	12.4832	0	0	6.6852	7.7151	0	0	**6.0615**	**6.2865**	0	0
$l = 30$	7.4228	9.4702	0	0	5.8306	6.3257	0	0	**5.0096**	**5.1784**	0	0
$l = 35$	7.6439	8.9241	0	0	4.6379	5.2644	0	0	**4.2346**	**4.3846**	0	0
$l = 40$	5.7481	7.8039	0	0	4.2027	4.6988	0	0	**3.4546**	**3.5451**	0	0
$l = 45$	4.9654	7.1958	0	0	3.0661	3.6264	0	0	**2.8071**	**2.8948**	0	0
$l = 50$	5.8373	7.4608	0	0	2.6048	2.8488	0	0	**2.2841**	**2.5016**	0	0

TABLE 5
Comparison of NSGA-II, NSCDE, RP-NSGA and RP-NSCDE for the identification of controlling regions of the neuronal network of cats' brain when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50. The mean and minimum values of f_1 (f_2) and their corresponding f_2 (f_1) in 20 runs are shown here. The best results among NSGA-II, NSCDE, RP-NSGA and RP-NSCDE are shown in **Bold fonts**.

	NSGA-II							NSCDE						
	Minimization f_1				Minimization f_2			Minimization f_1				Minimization f_2		
	f_1		f_2		f_1		f_2	f_1		f_2		f_1		f_2
	min	mean	min	mean	min	mean	mean	min	mean	min	mean	min	mean	mean
$l = 5$	33.1077	33.9679	0.0218	0.2779	33.2411	35.8399	0	33.1025	33.2071	0.0003	0.2851	33.1026	34.5485	0
$l = 10$	17.8696	19.0527	0.0772	0.5128	18.6041	19.6293	0	16.487	17.0332	0.3675	0.6336	16.8326	17.5784	0
$l = 15$	11.6244	12.9182	0.1139	0.4146	12.6848	13.3971	0	10.6783	11.3544	0.1047	0.4968	10.9391	11.6235	0
$l = 20$	8.3636	9.33	0.2123	0.4346	8.5891	9.5615	0	7.9199	8.4955	0.141	0.402	8.1125	8.5965	0
$l = 25$	6.4045	7.1511	0.245	0.4817	6.9028	7.3654	0	6.2831	6.7111	0.1133	0.3789	6.3769	6.7733	0
$l = 30$	5.5411	5.9728	0.3058	0.5445	5.5668	6.0746	0	5.0416	5.5146	0.3201	0.563	5.1696	5.5732	0
$l = 35$	4.4242	4.7877	0.2462	0.6071	4.5574	4.9243	0	4.3142	4.5908	0.2579	0.4623	4.3693	4.6429	0
$l = 40$	3.8775	4.0977	0	0.4077	3.9766	4.2008	0	3.4715	3.866	0.2656	0.5201	3.5475	3.9469	0
$l = 45$	3.1488	3.4535	0.3866	0.5959	3.2288	3.5923	0	2.88	3.2804	0.3357	0.5353	2.975	3.3153	0
$l = 50$	2.4316	2.8138	0.3484	0.6649	2.5796	2.9555	0	2.4215	2.7304	0.3074	0.5243	2.5005	2.7843	0

	RP-NSGA							RP-NSCDE						
	Minimization f_1				Minimization f_2			Minimization f_1				Minimization f_2		
	f_1		f_2		f_1		f_2	f_1		f_2		f_1		f_2
	min	mean	min	mean	min	mean	mean	min	mean	min	mean	min	mean	mean
$l = 5$	33.0986	33.2164	0.4631	0.4985	33.225	34.9851	0	**33.0924**	**33.1132**	0.0004	0.2813	**33.1018**	**33.5641**	0
$l = 10$	**16.4456**	16.563	0.4477	0.7878	17.321	18.9881	0	16.4717	**16.5113**	0.4315	0.7962	**16.7009**	**17.1039**	0
$l = 15$	10.4513	10.7292	0.5213	0.644	11.0029	12.0463	0	**10.3936**	**10.7217**	0.3543	0.7012	**10.7954**	**11.0937**	0
$l = 20$	7.6462	7.8904	0.334	0.5246	8.3936	8.8844	0	**7.6431**	**7.8189**	0.3141	0.4374	**7.7681**	**8.0189**	0
$l = 25$	6.0955	6.2613	0.5732	0.8094	6.5607	6.9731	0	**5.8792**	**6.166**	0.335	0.5735	**6.0615**	**6.2865**	0
$l = 30$	**4.8361**	**5.0419**	0.3595	0.7682	5.1509	5.5675	0	4.8831	5.0506	0.2203	0.4894	**5.0096**	**5.1784**	0
$l = 35$	**4.0954**	**4.1553**	0.7387	0.9598	4.4196	4.8480	0	4.1564	4.1999	0.4999	0.711	**4.2346**	**4.3846**	0
$l = 40$	**3.3423**	**3.3841**	0.4682	1.1474	3.6668	4.0162	0	3.3585	3.4044	0.5755	0.7588	**3.4546**	**3.5451**	0
$l = 45$	2.6894	2.7179	0.722	1.0586	3.1248	3.3405	0	**2.6884**	**2.7106**	0.7132	0.8654	**2.8071**	**2.8948**	0
$l = 50$	2.1745	**2.1902**	1.0415	1.3312	2.4749	2.9014	0	**2.1743**	2.1906	0.6895	0.9101	**2.2841**	**2.5016**	0

optimization method.

In summary, when only minimizing f_1, RP-NSCDE is comparable to JaDE. When only minimizing f_2, RP-NSCDE offers a much better performance than JaDE. Therefore, RP-NSCDE is more efficient and powerful than JaDE for the identification of controlling regions in the neuronal network. Hence, our results in this paper outperform the works in [9].

3.3.3 *Comparison of DyHF, IDyHF and RP-NSCDE*

In this subsection, we compare the performance of DyHF [59], IDyHF [38] and RP-NSCDE in this paper. Since DyHF and IDyHF are constraint optimization approaches, we introduce the following measure to characterize the feasibility of the solutions:

$$
\begin{aligned}
&Q = f_2 - \delta, \\
&H = f_2 - \delta, \\
&\tilde{f}_2(x) = \begin{cases} \max\{0, Q\}, & \text{if } \delta = 0.1, \\ \max\{0, |H|\}, & \text{if } \delta = 0. \end{cases}
\end{aligned} \tag{12}
$$

Then, $\tilde{f}_2(x)$ reflects the degree of constraint violation of the vector x. If $\tilde{f}_2(x) = 0$, the solution x is feasible.

The comparison results between DyHF, IDyHF and RP-NSCDE are given in Table 4 under different l and δ. It can be seen that DyHF, IDyHF and RP-NSCDE can obtain feasible solutions in each run among 20 runs and different dimension sizes. IDyHF improves DyHF due to the introduction of jDE to flexibly adjust the parameters of the population. RP-NSCDE performs best among the three algorithms in all the dimension sizes when $\delta = 0$ and $\delta = 0.1$, which validates the reliability and advantage of RP-NSCDE. Consequently, RP-NSCDE is more powerful and reliable than DyHF and IDyHF for the identification of controlling regions in the neuronal network. Consequently, our results work better than the works in [38].

3.3.4 *Comparison of NSGA-II, NSCDE, RP-NSGA and RP-NSCDE*

In RP-NSCDE, two important modifications are made including a reference-point strategy and the replacement of the search engine in NSGA-II by CoDE. Therefore, in this subsection, we compare NSGA-II, nondominated sorting composite differential evolution (NSCDE, i. e., RP-NSCDE without the reference point strategy), reference point-based nondominated sorting genetic algorithms (RP-NSGA, i. e., NSGA-II with the reference point strategy) and RP-NSCDE. The comparison results of NSGA-II, NSCDE, RP-NSGA and RP-NSCDE are shown in Fig. 2 and Table 5. The mean and minimum values of the nondominated solutions with minimum f_1 (f_2) and their corresponding f_2 (f_1) are provided in 20 runs under different l. All the four MOEAs can find the solutions with $f_2 = 0$ in each single run among 20 runs, which means that all these methods are effective for the identification of controlling areas in the neuronal network. It can be seen from Fig. 2 that the final nondominated solutions generated by RP-NSCDE dominate those generated by NSGA-II and NSCDE. Most of the solutions generated by RP-NSCDE dominate those obtained by RP-NSGA. RP-NSGA performs a little bit better than NSCDE, which indicates that the reference-point method is more powerful than the replacement of a search engine in NSGA-II by CoDE. Both RP-NSGA and NSCDE outperform the original NSGA-II, which means that the two modifications of NSGA-II are important to enhance the search ability of NSGA-II. From Table 5, although RP-NSGA performs a little bit better than RP-NSCDE in some cases, RP-NSCDE is more reliable than RP-NSGA in most of the cases, especially considering the corresponding f_1 when minimizing f_2. Overall, RP-NSCDE is the most powerful MOEA among the four algorithms. RP-NSGA and NSCDE ranks the second place and the third place, respectively. NSGA-II is the last one. Therefore, RP-NSCDE outperforms other variants of RP-NSCDE. The pareto fronts (PFs) are clearly plotted in Figs. 2 and 3. In addition, it can be found from Fig. 3 that as l increases, the nondominated solutions by RP-NSCDE become better, which manifests that the more driver nodes are allowed to control, the better the controllability is. Consequently, although NSGA-II, NSCDE, RP-NSGA and RP-NSCDE are effective for the identification of controlling areas in the neuronal network, the proposed RP-NSCDE offers the best performance among the four kinds of MOEAs due to the introduction of a reference point approach to guide the population to promising areas efficiently and CoDE to explore the search space by maintaining the diversity of the population.

3.4 Summarizing controlling regions by the nondominated solutions using RP-NSCDE

As demonstrated in Sec. 3.3, it is shown that RP-NSCDE has the promising ability of dealing with the identification of controlling regions in the neuronal network, in comparison with statistical methods [1], [9], JaDE [48], DyHF [59], IDyHF [38] and NSGA-II [45]. Therefore, we utilize RP-NSCDE to identify controlling areas of the neuronal network. Let η_i be the sum of each node to work as driver nodes in all the nondominated solutions generated by RP-NSCDE when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50. Table 6 presents the times of each node to be selected as driver nodes and its corresponding community. The regions with a large η_i can be viewed as controlling areas and it is found that these controlling areas are uniformly spread in the four communities. Table 6 shows that the regions such as VPc, 2 and 21a are the most important three regions to control the neuronal network, which spread evenly in three communities of the neuronal network (visual, auditory, somatomotor). For these regions, the importance of them

TABLE 6
Controlling regions by summarizing the nondominated solutions when $l = 5, 10, 15, 20, 25, 30, 35, 40, 45$ and 50.

Node Name	η	Community	$\triangle k$
VPc	1066	Auditory	4
2	1021	Somato-motor	7
21a	981	Visual	5
PS	897	Visual	7
AMLS	895	Visual	7
ALLS	895	Visual	4
21b	895	Visual	4
AAF	895	Auditory	3
Sb	895	Frontolimbic	8
Hipp	895	Frontolimbic	2
3a	828	Somato-motor	2
Tem	810	Auditory	2
SIV	810	Somato-motor	5
4	810	Somato-motor	3
PLLS	796	Visual	5
DLS	735	Visual	1
SII	731	Somato-motor	3
P	716	Auditory	3
RS	716	Frontolimbic	-2
PSb	716	Frontolimbic	3
1	713	Somato-motor	5
17	616	Visual	1
20b	616	Visual	0
PFCMiI	616	Frontolimbic	-3
19	583	Visual	3
Enr	574	Frontolimbic	-1
VLS	516	Visual	-2
AII	516	Auditory	1
SSAo	516	Somato-motor	-5
61	489	Somato-motor	0
PMLS	481	Visual	2
36	458	Frontolimbic	9
18	433	Visual	2
AI	403	Auditory	-1
PFCI	372	Frontolimbic	-10
4g	352	Somato-motor	-1
3b	345	Somato-motor	1
SSAi	314	Somato-motor	-5
Ig	314	Frontolimbic	5
7	294	Visual	-1
6m	294	Somato-motor	-4
5Bm	294	Somato-motor	-6
AES	216	Visual	-1
PFCMd	216	Frontolimbic	-6
Ia	192	Frontolimbic	-3
EPp	130	Auditory	-6
5Am	106	Somato-motor	-8
5BI	106	Somato-motor	-10
Cga	106	Frontolimbic	-13
35	106	Frontolimbic	7
20a	0	Visual	-6
5AI	0	Somato-motor	-10
CGp	0	Frontolimbic	-10

in neuroscience can be explained as follows: VPc is a tonotopical organized 'core' field in the auditory community; area 2 is the primary somatosensory area forming the periphery of the somato-motor community and 21a is the key area for hierarchical organization [52]. Additionally, the areas like 20a, 5AI and CGp are never to be selected to serve as controlling areas in all the nondominated solutions. Table 6 also illustrates $\triangle k = k_{in} - k_{out}$ of each node in the neuronal network. It can be inferred that the controlling areas are usually selected from the nodes with a large k_{in} and a small k_{out}. As discussed in [9], when l is small, the high-degree regions are important to control the entire networks. However, overall, not the same as the usual hubs detected by degree, BC and motif based methods in [41], the controlling areas found in this paper are usually selected from areas with a small degree rather than those with a large degree, which is consistent with the observations for the neuronal network of Caenorhabditis elegans in [29]. Most of the observations in this paper coincide with the findings in [9], [38].

4 DISCUSSION AND CONCLUSION

In this paper, the problem of multiobjective identification of controlling areas is investigated for the neuronal network of cats' brain. By treating two measures equally, the problem of multiobjective identification of controlling areas is formulated as a multiobjective optimization problem. By means of a multiobjective optimization algorithm, i. e., a reference point based nondominated sorting composite differential evolution (RP-NSCDE), is developed to tackle the proposed problem. By comparing with the methods and results in [9], [38], we show that RP-NSCDE is reliable and competitive to identify the controlling areas of the neuronal network and the nondominated solutions are shown in the form of pareto fronts (PFs). The findings reveal that the controlling areas are usually selected from the areas with a small degree, which differ from the usual hubs with a large degree in [41] and support the findings the results in [9], [38].

Finally, some future research topics are given here. Firstly, it remains interesting to use the information of network structure to initialize the population intentionally instead of randomly to achieve better performance. Secondly, it is important to extend our results in formation control of multi-agent systems. Thirdly, it is also interesting to utilize other kinds of multi-objective optimization methods to identify the controlling regions of neuronal networks [61], [62].

5 ACKNOWLEDGEMENTS

This research is supported by 973 Project (2009CB320600), the National Natural Science Foundation of China (60825303, 61021002, 61203235) and the Key Laboratory of Integrated Automation for the Process Industry (Northeastern University), SUMO (EU), IRTG 1740 (DFG), grant of the German Research Foundation (DFG) in the Research Group for 868 Computational Modeling of Behavioral, Cognitive, and Neural Dynamics and the Alexander von Humboldt Foundation of Germany.

REFERENCES

[1] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, pp. 47–97, 2002.

[2] M. J. Keeling and K. T. D. Eames, "Networks and epidemic models," *Journal of the Royal Society Interface*, vol. 2, pp. 295–307, 2005.

[3] D. Yu and U. Parlitz, "Inferring network connectivity by delayed feedback control," *PLoS ONE*, vol. 6, p. e24333, 2011.

[4] X. Qian and E. Dougherty, "Intervention in gene regulatory networks via phenotypically constrained control policies based on long-run behavior," *IEEE-ACM Trans. on Computational Biology and Bioinformatics*, vol. 9, pp. 123–136, 2012.

[5] S. Mitra, R. Das, and Y. Hayashi, "Genetic networks and soft computing," *IEEE-ACM Trans. on Computational Biology and Bioinformatics*, vol. 8, pp. 94–107, 2011.

[6] Z. Wang, X. Liu, Y. Liu, J. Liang, and V. Vinciotti, "An extended kalman filtering approach to modeling nonlinear dynamic gene regulatory networks via short gene expression time series," *IEEE-ACM Trans. on Computational Biology and Bioinformatics*, vol. 6, pp. 410–419, 2009.

[7] P. Li and J. Lam, "Disturbance analysis of nonlinear differential equation models of genetic sum regulatory networks," *IEEE-ACM Trans. Computational Biology and Bioinformatics*, vol. 8, pp. 253–259, 2011.

[8] E. Bullmore and O. Sporns, "The economy of brain network organization," *Nature Review Neuroscince*, vol. 13, pp. 336–349, 2012.

[9] Y. Tang, H. Gao, W. Zou, and J. Kurths, "Identifying controlling nodes in neuronal networks in different scales," *PLoS ONE*, vol. 7, p. e41375, 2012.

[10] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems." *Nature Review Neuroscince*, vol. 10, p. 1, 2009.

[11] P. E. Vértes, A. F. Alexander-Blocha, N. Gogtay, J. N. Giedd, J. L. Rapoport, and E. T. Bullmorea, "Simple models of human brain functional networks," *Proc. Natl Acad. Sci. USA*, vol. 109, pp. 5868–5873, 2012.

[12] G. Zamora-Lopez, E. Russo, P. M. Gleiser, C. S. Zhou, and J. Kurths, "Characterizing the complexity of brain and mind networks," *Philosophical Trans. R Soc. London A*, vol. 369, pp. 3730–3747, 2011.

[13] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge University Press, Cambridge, England, 2001.

[14] A. Arenas, A. Guilera, J. Kurths, Y. Moreno, and C. Zhou, "Synchronization in complex networks," *Physics Reports*, vol. 469, pp. 93–153, 2008.

[15] S. Boccaletti, J. Kurths, G. Osipov, D. L. Valladares, and C. S. Zhou, "The synchronization of chaotic systems," *Physics Reports*, vol. 366, pp. 1–101, 2002.

[16] P. Menck, J. Heitzig, N. Marwan, and J. Kurths, "How basin stability complements the linear-stability paradigm," *Nature Physics*, vol. 9, pp. 89–92, 2013.

[17] T. Vicsek and A. Zafeiris, "Collective motion," *Physics Reports*, vol. 517, pp. 71–140, 2012.

[18] Y. Tang and W. K. Wong, "Distributed synchronization of coupled neural networks via randomly occurring control," *IEEE Trans. On Neural Networks and Learning Systems*, vol. 24, pp. 435–447, 2013.

[19] H. Karimi and H. Gao, "New delay-dependent exponential h_∞ synchronization for uncertain neural networks with mixed time delays." *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 40, pp. 173–185, 2010.

[20] E. L. Lameu, C. A. S. Batista, A. M. Batista, K. Iarosz, R. L. Viana, S. R. Lopes, and J. Kurths, "Suppression of bursting synchronization in clustered scale-free (rich-club) neuronal networks," *CHAOS*, vol. 22, p. 043149, 2012.

[21] J. M. Palva, S. Monto, S. Kulashekhar, and S. Palva, "Neuronal synchrony reveals working memory networks and predicts individual memory capacity," *Proc. Natl Acad. Sci. USA*, vol. 107, pp. 7580–7585, 2010.

[22] A. Engel, P. Fries, and W. Singer, "Rapid feature selective neuronal synchronization through correlated latency shifting," *Nat Rev Neurosc*, vol. 2, pp. 704–716, 2001.

[23] P. Uhlhaas and W. Singer, "Neural synchrony in brain disorders: Relevance for cognitive dysfunctions and pathophysiology," *Neuron*, vol. 52, pp. 155–168, 2006.

[24] N. Kopell and B. Ermentrout, "Chemical and electrical synapses perform complementary roles in the synchronization of interneuronal networks," *Proc. Natl Acad. Sci. USA*, vol. 101, pp. 15 482–15 487, 2004.

[25] C. Zhou, L. Zemanová, G. Zamora, C. Hilgetag, and J. Kurths, "Hierarchical organization unveiled by functional connectivity in complex brain networks," *Phys Rev Lett*, vol. 97, p. 238103, 2006.

[26] X. Li, X. Wang, and G. Chen, "Pinning a complex dynamical network to its equilibrium," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 51, pp. 2074–2087, 2004.

[27] Y. Tang, Z. D. Wang, and J. Fang, "Pinning control of fractional-order weighted complex networks," *CHAOS*, vol. 19, p. 013112, 2009.

[28] W. Yu, G. Chen, and J. Lu, "On pinning synchronization of complex dynamical networks," *Automatica*, vol. 45, pp. 429–435, 2009.

[29] Y. Liu, J. Slotine, and A. Barabasi, "Controllability of complex networks," *Nature*, vol. 473, pp. 167–173, 2011.

[30] X. Wang, X. Li, and J. Lu, "Control and flocking of networked systems via pinning," *IEEE Circuits Syst. Mag.*, vol. 10, pp. 83–91, 2010.

[31] W. Yu, P. Lellis, G. Chen, M. Bernardo, and J. Kurths, "Distributed adaptive control of synchronization in complex networks," *IEEE Trans. On Automatic Control*, vol. 57, pp. 2153–2158, 2012.

[32] J. Lu, D. Ho, and Z. Wang, "Pinning stabilization of linearly coupled stochastic neural networks via minimum number of controllers," *IEEE Trans. Neural Networks*, vol. 20, pp. 1617–1629, 2010.

[33] F. Sorrentino, M. Bernardo, F. Garofalo, and G. Chen, "Controllability of complex networks via pinning," *Phys. Rev. E*, vol. 75, p. 046103, 2007.

[34] D. U. Hwang, M. Chavez, A. Amann, and S. Boccaletti, "Synchronization in complex networks with age ordering," *Phys. Rev. Lett.*, vol. 94, p. 138701, 2005.

[35] J. Lu, J. Kurths, J. Cao, N. Mahdavi, and C. Huang, "Synchronization control for nonlinear stochastic dynamical networks: Pinning impulsive strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, pp. 285–292, 2012.

[36] Y. Tang, H. Gao, J. Kurths, and J. Fang, "Evolutionary pinning control and its application in UAV coordination," *IEEE Trans. On Industrial Informatics*, vol. 8, pp. 828–838, 2012.

[37] M. Egerstedt, "Degrees of control," *Nature*, vol. 473, pp. 158–159, 2011.

[38] Y. Tang, Z. Wang, H. Gao, S. Swift, and J. Kurths, "A constrained evolutionary computation method for detecting controlling regions of cortical networks," *IEEE-ACM Trans. On Computational Biology and Bioinformatics*, vol. 9, pp. 1569–1581, 2012.

[39] P. A. Robinson, J. A. Henderson, E. Matar, P. Riley, and R. T. Gray, "Dynamical reconnection and stability constraints on cortical network architecture," *Phys. Rev. Lett.*, vol. 103, p. 108104, 2009.

[40] J. Gomez, G. Zamora, Y. Moreno, and A. Arenas, "From modular to centralized organization of synchronization in functional areas of the cat cerebral cortex," *PLoS ONE*, vol. 5, p. e12313, 2010.

[41] O. Sporns, C. Honey, and R. Kotter, "Identification and classification of hubs in brain networks," *PLoS ONE*, vol. 10, p. e1049, 2007.

[42] L. Z. Liu, F. X. Wu, and W. J. Zhang, "Inference of biological s-system using separable estimation method and genetic algorithm," *IEEE-ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, pp. 955–965, 2012.

[43] W. Zhu, J. Fang, Y. Tang, W. Zhang, and W. Du, "Digital iir filters design using differential evolution algorithm with a controllable probabilistic population size," *PLoS ONE*, vol. 7, p. e40549, 2012.

[44] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley., 2002.

[45] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. On Evolutionary Computation*, vol. 6, pp. 182–197, 2002.

[46] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm," *TIK-Report 103*, 2001.

[47] K. Deb, A. Sinha, P. Korhonen, and J. Wallenius, "An interactive evolutionary multi-objective optimization method based on progressively approximated value functions," *KanGAL Report No. 2009005*, 2009.

[48] J. Q. Zhang and A. C. Sanderson, "Jade: adaptive differential evolution with optional external archive," *IEEE Trans. on Evolutionary Computation*, vol. 13, pp. 945–958, 2009.

[49] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Trans. on Evolutionary Computation*, vol. 10, pp. 646–657, 2006.

[50] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, pp. 4–31, 2011.

[51] Y. Wang, Z. Cai, and Q. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Trans. On Evolutionary Computation*, vol. 15, pp. 55–66, 2011.

[52] J. Scannell, G. Burns, C. Hilgetag, M. O'Neill, and M. Young, "The connectional organization of the cortico-thalamic system of the cat," *Cereb Cortex*, vol. 9, pp. 277–299, 1999.

[53] C. Hilgetag, G. Burns, M. O'Neill, J. Scannell, and M. Young, "Anatomical connectivity defines the organization of clusters of cortical areas in the macaque monkey and the cat," *Phil Trans R Soc London B*, vol. 355, pp. 91–110, 2000.

[54] G. Zamora, C. Zhou, and J. Kurths, "Cortical hubs form a module for multisensory integration on top of the hierarchy of cortical networks," *Frontiers in Neuroinformatics*, vol. 4, pp. 1–12, 2010.

[55] C. Zhou, L. Zemanová, C. Hilgetag, and J. Kurths, "Structure-function relationship in complex brain networks expressed by hierarchical synchronization," *New J Phys*, vol. 9, p. 178, 2007.

[56] L. Zemanova, C. Zhou, and J. Kurths, "Structural and functional clusters of complex brain networks," *Physica D*, vol. 224, pp. 202–212, 2006.

[57] Y. Tang, Z. Wang, W. K. Wong, J. Kurths, and J. Fang, "Multiobjective synchronization of coupled systems," *CHAOS*, vol. 21, p. 025114, 2011.

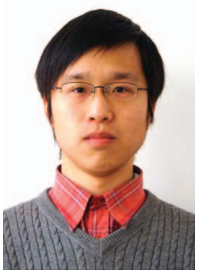
[58] K. Deb and N. Padhye, "Improving a particle swarm optimization algorithm using an evolutionary algorithm framework, KanGAL No. 2010003," 2010.

[59] Y. Wang and Z. Cai, "A dynamic hybrid framework for constrained evolutionary optimization," *IEEE Trans. Systems, man, and cybernetics - Part B: Cybernetics*, vol. 42, pp. 203–217, 2012.

[60] X. Wang and G. Chen, "Synchronization in scale-free dynamical networks: Robustness and fragility," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 49, pp. 54–62, 2002.

[61] S. Loghmanian, R. Yusof, M. Khalid, and F. Ismail, "Polynomial narx model structure optimization using multi-objective genetic algorithm," *Int. J. of Innovative Computing Information and Control*, vol. 8, pp. 7341–7362, 2012.

[62] M. Jahromi, M. Ehsan, A. Meyabadi, and T. Niknam, "An interactive fuzzy multi-objective approach for short term dg planning," *Int. J. of Innovative Computing Information and Control*, vol. 8, pp. 4157–4175, 2012.



Yang Tang (M'11) received the B. S. degree and PhD degree in electrical engineering from Donghua University, Shanghai, China in 2006 and 2011, respectively. From December 2008 to December 2010, he was a research associate in The Hong Kong Polytechnic University, Hung Hom Kowloon, Hong Kong, China. Now he is an Alexander von Humboldt research fellow at Humboldt University of Berlin, Berlin, Germany and a visiting scientist at Potsdam Institute for Climate Impact Research, Potsdam, Germany since 2011. He was a visiting research fellow at Brunel University in the UK from May 2012-June 2012. He has published more than 30 refereed papers in international journals. His main research interests are synchronization/consensus, networked control system, evolutionary computation, bioinformatics and their applications. He is a very active reviewer for many international journals. He is a member of the IEEE.

Huijun Gao (SM'09) received the Ph.D. degree in control science and engineering from Harbin Institute of Technology, China, in 2005. He was a Research Associate with the Department of Mechanical Engineering, The University of Hong Kong, from November 2003 to August 2004. From October 2005 to October 2007, he carried out his postdoctoral research with the Department of Electrical and Computer Engineering, University of Alberta, Canada. Since November 2004, he has been with Harbin Institute of Technology, where he is currently a Professor and director of the Research Institute of Intelligent Control and Systems. Dr Gao's research interests include network-based control, robust control/filter theory, time-delay systems and their engineering applications. He is an Associate Editor for Automatica, IEEE Transactions on Industrial Electronics, IEEE Transactions on Systems Man and Cybernetics Part B: Cybernetics, IEEE Transactions on Fuzzy Systems, IEEE Transactions on Circuits and Systems - I, IEEE Transactions on Control Systems Technology etc. He is a senior member of the IEEE.

Jürgen Kurths studied mathematics at the University of Rostock and got his PhD in 1983 at the GDR Academy of Sciences. He was full Professor at the University of Potsdam from 1994-2008 and has been Professor of Nonlinear Dynamics at the Humboldt University, Berlin and chair of the research domain Transdisciplinary Concepts of the Potsdam Institute for Climate Impact Research since 2008 and a 6th century chair of Aberdeen University (UK) since 2009. He is a fellow of the American Physical Society. He got an Alexander von Humboldt research award from CSIR (India) in 2005 and a Honory Doctorate in 2008 from the Lobachevsky University Nizhny Novgorod and one in 2012 from the State University Saratov. He has become a member of the Academia Europaea in 2010 and of the Macedonian Academy of Sciences and Arts in 2012. His main research interests are synchronization, complex networks, time series analysis and their applications. He has published more than 500 papers which are cited more than 18,000 times (H-factor: 56). He is an Editor for PLoS ONE, Philosophical Transaction of The Royal Society A, Journal of Nonlinear Science, CHAOS etc.